

# 0. Base e Conceitos

## Regiões, AZs, Zonas Locais, Edge Locations e etc

### Regiões

Conjunto de Data Centers

Cada região está separada das outras

Projetado para funcionar independentemente de outra região

Existe diferença do valor cobrado para o mesmo serviço entre as regiões

A região Norte da Virgínia (us-east-1) é onde os serviços chegam primeiro e normalmente mais barato

No console da AWS é disponibilizado somente os recursos que estão vinculados à região especificada

Os recursos não são replicados entre regiões da AWS automaticamente

Os custos são disponibilizados por região, caso exista mais de uma região sendo usada

Cada região possui pelo menos 2 Zonas de Disponibilidade

### Zonas de Disponibilidade

Uma AZ é um conjunto de data centers (Local Zones), sendo redundante e seguro

Projetado para trabalhar de forma isolada das demais AZs

Haverá um balanceamento de carga entre as AZs

Um Datacenter não necessariamente será uma AZ, podendo haver vários datacenters para constituir uma AZ

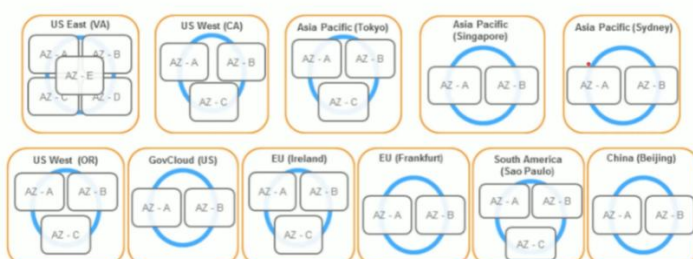
Cada região possui pelo menos 2 Zonas de Disponibilidade

A região física é AZ e o conjunto delas é uma região, ex: região de São Paulo com 3 AZs

As AZ ficam geograficamente distantes umas das outras para que, caso ocorra uma catástrofe, todas não sejam afetadas

Elas são construídas a uma distância que possibilite uma interconexão de rede de baixa latência

Exemplo de Regiões e Zonas de Disponibilidade:



Existem pelo menos 2 AZ's em cada Região

### AWS Edge Locations

Os locais de extremidade são pontos de extremidade para AWS que são usados para conteúdo em cache e usados como rede de entrega de conteúdo (CDN). Isso consiste no Amazon Cloud Front (CF).

Há muito mais pontos de presença do que regiões (217 pontos de presença (205 pontos de presença e 12 caches de borda regionais)) em todo o mundo. Os locais de borda atendem a solicitações para CloudFront e Route 53

- Caches do CloudFront
- Serviços de CDN
- Route 53 (Serviço de DNS)
- Amazon S3 Transfer Acceleration

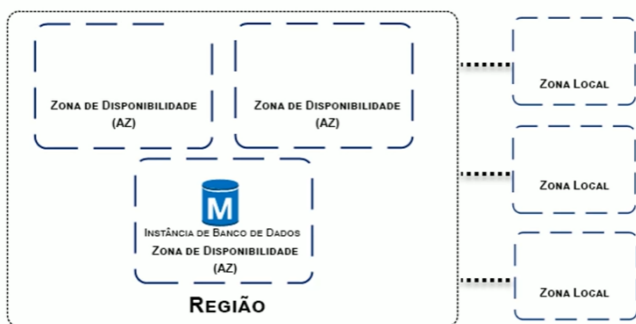
CloudFront é uma rede de entrega de conteúdo, enquanto Route 53 é um serviço DNS. As solicitações enviadas para qualquer um desses serviços serão roteadas para o ponto de presença mais próximo automaticamente. **Isso permite baixa latência, não importa onde o usuário final esteja localizado**

### AWS Local Zones

As zonas locais da AWS permitem que você use serviços selecionados da AWS, como serviços de computação e armazenamento, mais perto de mais usuários finais, fornecendo-lhes acesso de latência muito baixa aos aplicativos executados localmente

As zonas locais da AWS também estão conectadas à região pai por meio da rede privada redundante e de largura de banda muito alta da Amazon, oferecendo aos aplicativos em execução nas zonas locais da AWS acesso rápido, seguro e contínuo aos demais serviços da AWS

As zonas locais da AWS têm sua própria conexão com a Internet e oferecem suporte ao AWS Direct Connect, de modo que os recursos criados na zona local podem servir aos usuários finais locais com comunicações de latência muito baixa



### Wavelength (infraestrutura para aplicações 5G)

Forneça aplicações de latência ultrabaixa para dispositivos 5G. O AWS Wavelength é uma oferta de infraestrutura da AWS otimizada para aplicações de computação de borda móveis.

As Zonas do Wavelength são implantações de infraestrutura da AWS que incorporam serviços de computação e armazenamento da AWS aos datacenters dos provedores de serviços de comunicação (CSP) na borda da rede 5G, para que o tráfego de aplicativos de dispositivos 5G possa acessar os servidores de aplicativos em execução nas Zonas do Wavelength sem sair da rede de telecomunicações.

Isso evita que a latência, que resultaria do tráfego de aplicações, tenha que atravessar vários saltos na Internet para chegar ao seu destino, permitindo que os clientes aproveitem ao máximo os benefícios de latência e largura de banda oferecidos pelas redes 5G modernas.

Como funciona: É fácil começar. Basta fazer login no Console de Gerenciamento da AWS e habilitar as Zonas do Wavelength que você deseja usar para sua conta

Casos de Uso: O AWS Wavelength permite aplicativos 5G que oferecem experiências interativas e imersivas, como streaming de jogos, realidade virtual e experiências no local para eventos em tempo real

### Outposts (Nuvem híbrida)

São datacenters que tem um pouco da infraestrutura da AWS, ou seja, são instalados racks para disponibilizar os serviços da AWS. O AWS Outposts é um serviço gerenciado que oferece a mesma infraestrutura, os serviços, as APIs e as ferramentas da AWS a praticamente qualquer datacenter ou instalação on-premises para oferecer uma experiência híbrida verdadeiramente consistente

Benefícios: Com o AWS Outposts, você pode executar o Amazon EC2; o Amazon EBS; o Amazon S3; serviços baseados em contêiner, como o Amazon EKS, e serviços de banco de dados, como o Amazon RDS, no AWS Outposts e serviços de análises, como o Amazon EMR, no local. Você pode estender perfeitamente sua Amazon Virtual Private Cloud no local, executar alguns serviços da AWS localmente em Outposts e conectar-se a uma ampla variedade de serviços disponíveis na Região da AWS local

### Escopo de Serviços

É definido se o serviço será executado dentro de uma AZ, dentro de todas as AZ's ou se é executado globalmente, dentro de todas as regiões da AWS

Assim temos 3 níveis de escopo de serviço

1. Escopo de AZ
2. Escopo de Região
3. Escopo Global

#### **Escopo de AZ**

- EC2
- RDS
- EBS
- Private IP

#### **Escopo de Região**

- ELB
- AutoScaling
- AMI
- S3
- Security Group

#### **Escopo Global**

- IAM
- Route 53
- CloudFront

### **Links Úteis**

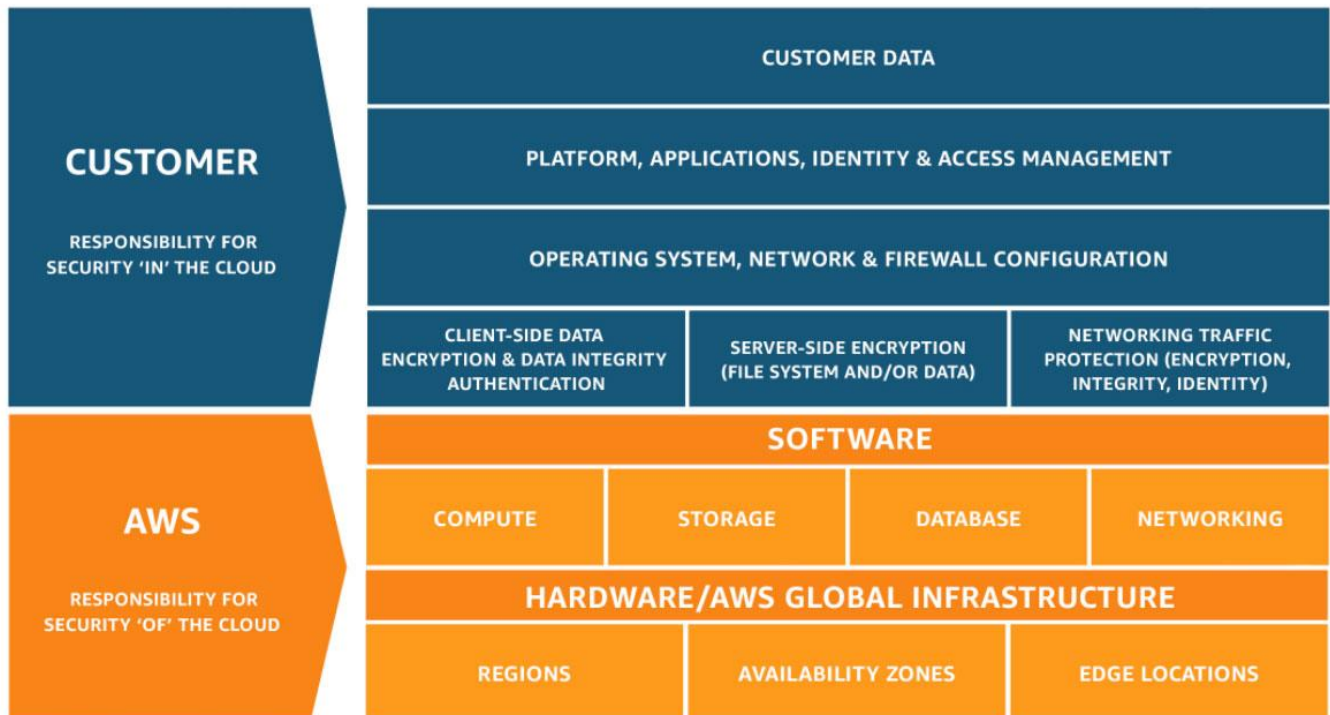
<https://docs.aws.amazon.com/general/latest/gr/aws-arns-and-namespaces.html>

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/resource-ids.html>

### **➤ Pontos de Atenção**

A company is using hundreds of AWS resources in multiple AWS regions. They require a way to uniquely identify all of their AWS resources that will allow them to specify a resource unambiguously across all of AWS, such as in IAM policies, Amazon Relational Database Service (Amazon RDS) tags, and API calls. Which of the following is the most suitable option to use in this scenario? R: Amazon Resource Name

# Modelo de Responsabilidade Compartilhada



# Nível Gratuito da AWS

O nível gratuito da AWS oferece a capacidade de explorar e testar gratuitamente serviços da AWS até os limites especificados para cada serviço

**Acessar o link:** <https://aws.amazon.com/pt/free/>

**Existem 3 tipos de gratuidade:**



## Sempre gratuito

Essas ofertas de nível gratuito não expiram e estão disponíveis para todos os clientes da AWS



## 12 meses gratuitos

Aproveite essas ofertas por 12 meses após a data de seu cadastramento inicial na AWS



## Testes

As ofertas de teste gratuito de curto prazo começam na data em que você ativa determinado serviço

### ☒ Suporte Basic - gratuito

- Recomendado para novos usuários que estão começando a usar a AWS
- Acesso por autoatendimento 24 horas por dia/7 dias por semana a todos os recursos da AWS
- Apenas para questões de conta e faturamento
- Acesso ao Personal Health Dashboard e ao Trusted Advisor



### ☐ Suporte Developer - a partir de 29 USD/mês

- Recomendado para desenvolvedores que estão avaliando a AWS
- Acesso por e-mail ao AWS Support durante o horário comercial
- Tempo de resposta de 12 horas (horário comercial)



### ☐ Suporte Business - a partir de 100 USD/mês

- Recomendado para executar cargas de trabalho de produção na AWS
- Suporte técnico 24 horas por dia/7 dias por semana por e-mail, telefone e chat
- Tempo de resposta de 1 hora
- Conjunto completo de recomendações de melhores práticas do Trusted Advisor



### Precisa de suporte de nível Enterprise?

A partir de 15.000 USD por mês, você receberá tempo de resposta de 15 minutos e experiência ao estilo de concierge com um Gerente técnico da conta designado. [Saiba mais](#)

# O que é Serverless?

Serverless (sem servidor) é uma **arquitetura nativa da nuvem orientada a eventos** e que permite transferir mais das suas responsabilidades operacionais à AWS

Ela permite criar e executar aplicativos e serviços sem se preocupar com a infraestrutura em que esses aplicativos estão rodando e elimina as tarefas de gerenciamento de servidores

Deixa de ser necessário tarefas como o provisionamento de servidores ou de clusters, patches, manutenção do sistema operacional e provisionamento de capacidade

Podemos criar servidores para praticamente qualquer tipo de aplicativo ou serviço de back-end, e a AWS cuida de tudo o que for necessário para executar e escalar aplicativos com alta disponibilidade, reduzindo custos

Permite focar mais no desenvolvimento da aplicação e menos em configuração ou gerenciamento de servidores

## Como funciona de fato?

### 1. Primeiro vamos relembrar a Arquitetura Padrão (tradicional):

Servidor 24h ligado mesmo sem usuários utilizando

Servidor está sempre ouvindo as requisições dos diferentes clients

Para escalar temos que subir mais máquinas

Mas sempre temos ao menos 1 máquina online para caso algum usuário acesse

Além destes pontos, precisamos estimar o fluxo de usuários e provisionar capacidade da máquina para receber esta demanda estimada

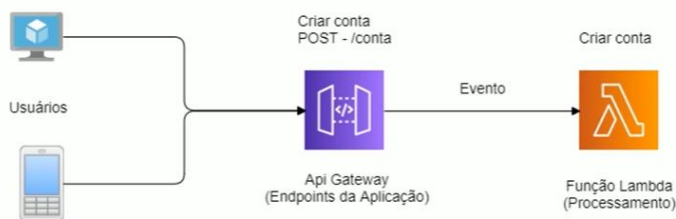
### 2. Já na Arquitetura Serverless (sem servidor):

Ainda existem servidores, porém eles funcionam de uma maneira um pouco diferente

Não existe um servidor online “ouvindo” as requisições, quando chega uma requisição a AWS sobe uma máquina para processá-la e depois a encerra

Ou seja, a partir de um Evento, uma Função Lambda é ativada para processar uma tarefa

Só paga pelo tempo em que a Função Lambda subiu, processou e devolveu a resposta, ou seja, só paga pelo tempo de processamento, ou seja, quanto mais tempo uma Função Lambda ficar processando mais caro fica



Exemplo 2: um arquivo chega no S3 Bucket, é possível programar eventos para serem disparados a acionar Funções Lambdas que importa o arquivo para Dynamo DB



Exemplo 3: suponha que todo dia ao meio dia aconteça um processamento, podemos usar o Cloud Watch para disparar um evento que executa uma Função Lambda



É claro que conforme a necessidade, como por exemplo várias requisições, diversas Funções Lambdas podem ficar ativas ao mesmo tempo

Ou seja, existem várias formas de se invocar uma Função Lambda, e ela faz tudo o que um servidor comum faz

Apenas devemos ter em mente que a máquina é mais limitada, e que a inicialização do código deve ser o mais rápido possível

Ou seja, as Funções Lambdas devem ser concisas e bem definidas, para que sua inicialização e processamento seja rápido e valha a pena

# AWS SAM (Serverless Application Model)

O AWS SAM ou Serverless Application Model é um framework open-source para construir aplicações Serverless na AWS. Fornece sintaxe abreviada para expressar funções, APIs, databases e mapeamentos de origens de eventos

É concorrente do Serverless framework, é considerado o framework oficial da AWS, ambos tendem a facilitar o desenvolvimento, criar o ambiente de forma simplificada gerando as aplicações através de template e deployando de forma muito mais fácil

Basicamente, através de template e linha de comando, conseguimos criar nossas aplicações Serverless, testar localmente simulando o ambiente da AWS e fazer deploy de forma simplificada

Totalmente gratuito, é integrado com ferramentas de desenvolvimento (ex: plugin no vscode AWS Toolkit), podendo executar o ambiente em containers do Docker, baixando as imagens direto da AWS. O único trabalho então é instalá-lo, executável pode ser baixado pelo link acima, após instalar o **aws\_sam\_cli**, executar o comando **sam --version**

```
#Step 1 - Download a sample application
```

```
sam init
```

```
#Step 2 - Build your application
```

```
cd sam-app
```

```
sam build
```

```
#Step 3 - Deploy your application
```

```
sam deploy --guided
```

Invocando função localmente:

<https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-using-invoke.html>

```
# Invoking function with event file
```

```
$ sam local invoke "HelloWorldFunction" -e events/event.json
```

```
# Invoking api gateway local
```

```
$ sam local start-api
```

Links Úteis:

<https://aws.amazon.com/pt/serverless/sam/>

<https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-getting-started-hello-world.html>



# AWS Step Function

O AWS Step Functions é um serviço de fluxo de trabalho visual que utiliza pouco código, usado para orquestrar serviços da AWS, automatizar processos de negócios e criar aplicações sem servidor. Os fluxos de trabalho gerenciam falhas, novas tentativas, paralelização, integrações de serviços e observabilidade, de modo que os desenvolvedores possam se concentrar na lógica de negócios com maior valor.

O AWS Step Functions é um orquestrador de funções sem servidor (Serverless) que facilita o sequenciamento de funções do AWS Lambda e vários serviços da AWS

Por meio da interface visual, é possível criar e executar uma série de fluxos de trabalho com ponto de verificação e orientados a eventos que mantêm o estado da aplicação

Por exemplo: fazer uma linha onde uma função executa, que manda o retorno dela para outra função, executa e manda o retorno para uma terceira função. É possível ter loops, uma função chamar ela mesma, pode haver também uma tomada de decisão, como por exemplo, se a execução vai para o fluxo A ou fluxo B, dentre outras opções. Ou seja, com Step Functions é possível ter um maior controle de processamento das funções

A saída de uma etapa serve de entrada para a próxima. Cada etapa da aplicação é executada em sequencia e conforme esperado pela lógica de negócios definida

O Step Functions gerencia automaticamente o tratamento de erros, a lógica de novas tentativas e o estado

## Quando Utilizar?

Processamento de dados: Ele ajuda a garantir que trabalhos de ETL (tarefas em background de Extrair Transformar e Carregar) múltiplos e prolongados sejam executados em ordem e concluídos com sucesso, em vez de garantir manualmente esses trabalhos ou manter um aplicativo separado

*ETL, do inglês Extract Transform Load (Extrair Transformar Carregar), são ferramentas de software cuja função é a extração de dados de diversos sistemas, transformação desses dados conforme regras de negócios e por fim o carregamento dos dados geralmente para um Data Mart e/ou Data Warehouse, porém nada impede que também seja para enviar os dados para um determinado sistema da organização. (Wikipedia)*

Também é possível usar o Step Function para padronizar um fluxo de trabalho de treinamento de Machine Learning para aumentar a precisão dos modelos de Machine Learning

Modernizar um monolito: O uso do Step Functions para destacar algumas tarefas do resto da base de código permite que você aborde a transformação de aplicativos monolíticos em microsserviços como uma série de pequenas etapas

Orquestração de aplicativos: Use o Step Functions para combinar várias funções do AWS Lambda em aplicativos e microsserviços sem servidor com alta capacidade de resposta, sem necessidade de criar código para lógica de fluxo de trabalho, processamentos paralelos, tratamento de erros, esgotamentos de tempos limite ou novas tentativas de execução

O Step Functions tem dois tipos de fluxos de trabalho: Standard Workflows e Express Workflows

## Definição de Preço

A cobrança é realizada por Transições de Estado, ou seja, cada mudança de Step (etapa do fluxo de trabalho) que é executado é contado como uma transição de estado. O Nível Gratuito da direito a 4000 transições de estado / mês para Step Functions Standard Workflows. Para tipo Express Workflows a cobrança ocorre no momento que acontece uma solicitação, ou seja, a cobrança é pelo número total de solicitação em todos os fluxos de trabalho, inclusive os testes no console são contados

## Links Úteis

<https://docs.aws.amazon.com/step-functions/latest/dg/amazon-states-language-choice-state.html>

<https://docs.aws.amazon.com/step-functions/latest/dg/sfn-stuck-execution.html>

<https://aws.amazon.com/step-functions/features/>

➤ **Pontos de Atenção**

A financial company instructed you to automate the recurring tasks in your department such as patch management, infrastructure selection, and data synchronization to improve their current processes. You need to have a service which can coordinate multiple AWS services into serverless workflows. Which of the following is the most cost-effective service to use in this scenario? R: AWS Step Functions

# AWS CodeCommit

Basicamente é um serviço de Git da própria AWS

Primeiramente, é necessário gerar as “Credenciais HTTPS do Git para o AWS CodeCommit” dentro das “Credenciais de Segurança” de um usuário ativo no IAM. Observação: não é possível configurar conexões SSH para uma conta raiz.

Próximo passo, no CodeCommit criar um repositório e copiar a URL para clonar o projeto

No momento em que o projeto for clonado localmente, o propt apresentará uma tela para informar as credenciais geradas no IAM

## **Definição de Preço**

Gratuito para os 5 usuários ativos, com repositório ilimitado, 50 GB de armazenamento por mês e 10.000 solicitações Git/mês (push/pull etc.)

# AWS CodeDeploy

## Working with deployment groups in CodeDeploy

You can specify one or more deployment groups for a CodeDeploy application. Each application deployment uses one of its deployment groups. The deployment group contains settings and configurations used during the deployment. Most deployment group settings depend on the compute platform used by your application. Some settings, such as rollbacks, triggers, and alarms can be configured for deployment groups for any compute platform.

### Deployment groups in Amazon ECS compute platform deployments

In an Amazon ECS deployment, a deployment group specifies the Amazon ECS service, load balancer, optional test listener, and two target groups. It also specifies when to reroute traffic to the replacement task set and when to terminate the original task set and Amazon ECS application after a successful deployment.

### Deployment groups in AWS Lambda compute platform deployments

In an AWS Lambda deployment, a deployment group defines a set of CodeDeploy configurations for future deployments of an AWS Lambda function. For example, the deployment group specifies how to route traffic to a new version of a Lambda function. It also might specify alarms and rollbacks. A single deployment in an AWS Lambda deployment group can override one or more group configurations.

### Deployment groups in EC2/On-Premises Compute Platform deployments

In an EC2/On-Premises deployment, a deployment group is a set of individual instances targeted for a deployment. A deployment group contains individually tagged instances, Amazon EC2 instances in Amazon EC2 Auto Scaling groups, or both.

In an in-place deployment, the instances in the deployment group are updated with the latest application revision.

In a blue/green deployment, traffic is rerouted from one set of instances to another by deregistering the original instances from a load balancer and registering a replacement set of instances that typically has the latest application revision already installed.

You can associate more than one deployment group with an application in CodeDeploy. This makes it possible to deploy an application revision to different sets of instances at different times. For example, you might use one deployment group to deploy an application revision to a set of instances tagged Test where you ensure the quality of the code. Next, you deploy the same application revision to a deployment group with instances tagged Staging for additional verification. Finally, when you are ready to release the latest application to customers, you deploy to a deployment group that includes instances tagged Production.

## Links Úteis

<https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-groups.html>

### ➤ Pontos de Atenção

A web application is hosted on a fleet of EC2 instances inside an Auto Scaling Group with a couple of Lambda functions for ad hoc processing. Whenever you release updates to your application every week, there are inconsistencies where some resources are not updated properly. You need a way to group the resources together and deploy the new version of your code consistently among the groups with minimal downtime. Which among these options should you do to satisfy the given requirement with the least effort? R: Use deployment groups in CodeDeploy to automate code deployments in a consistent manner.

# AWS CodePipeline

## ➤ Pontos de Atenção

Uma empresa de mídia global está usando o AWS CodePipeline como um serviço de automação para lançar novos recursos aos clientes. Todos os códigos são carregados no bucket do Amazon S3. As alterações nos arquivos armazenados no intervalo S3 devem acionar o AWS CodePipeline, que iniciará ainda mais o AWS Elastic Beanstalk para a implantação de recursos adicionais. Qual é o requisito adicional que deve ser configurado para acionar o CodePipeline de maneira mais rápida?

R: Disable periodic checks, create an Amazon CloudWatch Events rule & AWS CloudTrail trail.

Para acionar automaticamente o pipeline com mudanças no bucket S3 de origem, a regra de eventos do Amazon CloudWatch e a trilha do AWS CloudTrail devem ser aplicadas. Quando há uma mudança no intervalo S3, os eventos são filtrados usando o AWS CloudTrail e os eventos do Amazon CloudWatch são usados para acionar o início do pipeline. Este método padrão é mais rápido e as verificações periódicas devem ser desabilitadas para ter o acionamento baseado em eventos de CodePipeline. Você pode usar as seguintes ferramentas para monitorar seus pipelines CodePipeline e seus recursos:

- **Amazon CloudWatch Events** - Use os eventos do Amazon CloudWatch para detectar e reagir às mudanças de estado de execução do pipeline (por exemplo, enviar uma notificação do Amazon SNS ou invocar uma função Lambda).
- **AWS CloudTrail** - Use o CloudTrail para capturar chamadas de API feitas por ou em nome de CodePipeline em sua conta da AWS e entregar os arquivos de log a um bucket do Amazon S3. Você pode escolher que o CloudWatch publique notificações do Amazon SNS quando novos arquivos de log forem entregues para que você possa tomar medidas rápidas.
- **Console e CLI** - você pode usar o console CodePipeline e CLI para visualizar detalhes sobre o status de um pipeline ou uma execução de pipeline particular.

# AWS Cognito

O Amazon Cognito permite adicionar cadastramento, login e controle de acesso de usuários a aplicações Web e móveis. O Amazon Cognito pode ser escalado para milhões de usuários e oferece suporte a login com provedores de identidade social como Apple, Facebook, Google e Amazon e com provedores de identidade empresariais via SAML 2.0 e OpenID Connect.

## Trocando em miúdos

1. O usuário da aplicação faz login utilizando Facebook, Google ou qualquer outro provedor de identidade, porém quem por trás faz a requisição para o provedor é o User Pool do Cognito.
2. O Cognito devolve um token que logo em seguida a aplicação requisita ao Identity Pool os acessos permitidos para aquele usuário.
3. Tendo as credenciais necessárias, os serviços da AWS serão acessados

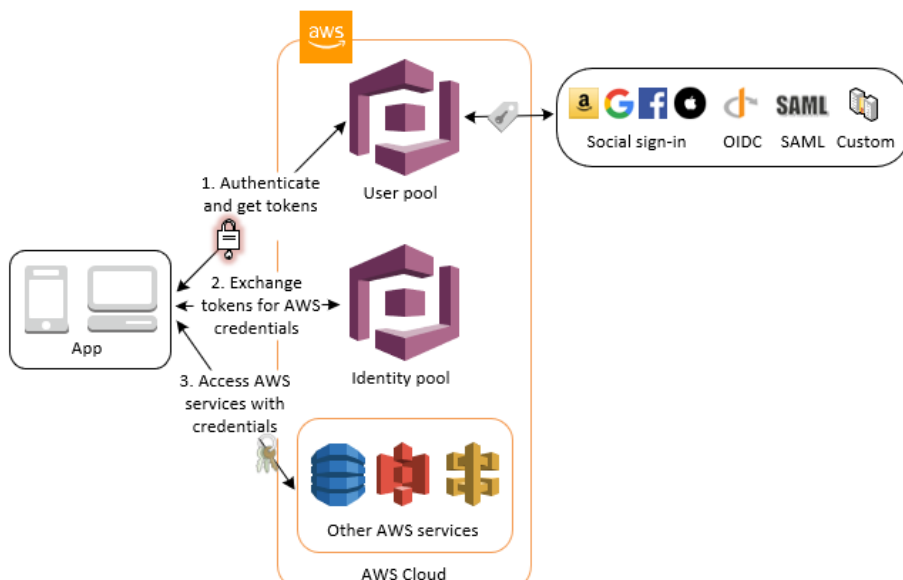
Observação: um tópico SNS pode ser criado para notificar outras aplicações de que aquele usuário está autenticado

## Unauthenticated Users

Amazon Cognito Identity Pools can support unauthenticated identities by providing a unique identifier and AWS credentials for users who do not authenticate with an identity provider. Unauthenticated users can be associated with a role with limited access to resources compared to a role for authenticated users.

## Links Úteis

- <https://aws.amazon.com/cognito/>
- <http://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-integrate-with-cognito.html>
- <https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-identity-pools.html>
- <https://docs.aws.amazon.com/cognito/latest/developerguide/identity-pools.html>
- <https://aws.amazon.com/cognito/getting-started/>
- <https://docs.aws.amazon.com/cognito/latest/developerguide/concepts.html>



## ➤ Pontos de Atenção

1. You have implemented AWS Cognito services to require users to sign in and sign up to your app through social identity providers like Facebook, Google, etc. Your marketing department wants users to anonymously try out the app because the current log-in requirement is excessive, which may reduce the demand for products and services offered through the app. What would you suggest to the marketing department in this regard? R: Cognito Identity supports guest users for the ability to enter the app and have limited access.

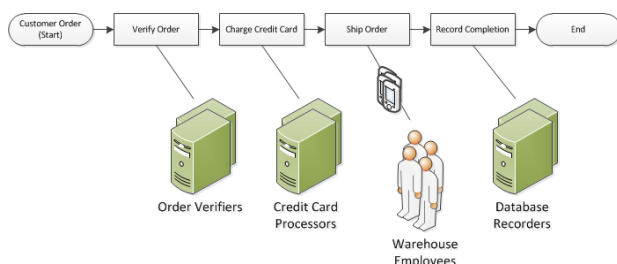
2. Você projetou um aplicativo que usa recursos da AWS, como o S3, para operar e armazenar os documentos dos usuários. Atualmente, você usa pools de identidade e pools de usuários do Cognito. Para aumentar o uso e a facilidade de inscrição, você decide que adicionar federação de identidade social é o melhor caminho a seguir. Como você diferenciaria o pool de identidade Cognito e os provedores de identidade federados (por exemplo, Google)? R: You can choose a federated identity provider to authenticate users and associate a Cognito identity pool to authorize the users.
3. Seu aplicativo usa AWS Cognito Identity para autenticação e armazena perfis de usuário em um pool de usuários. Para expandir a disponibilidade e facilidade de login no aplicativo, sua equipe está solicitando conselhos sobre como permitir o uso de provedores de identidade OpenID Connect (OIDC) como meio adicional de autenticar usuários e salvar as informações do perfil do usuário. Qual é a sua recomendação sobre provedores de identidade OIDC? R: This is supported, along with social and SAML based identity providers. OpenID Connect (OIDC) identity providers (IdPs) (like Salesforce or Ping Identity) are supported in Cognito, along with social and SAML based identity providers. You can add an OIDC IdP to your user pool in the AWS Management Console, with the AWS CLI, or with the user pool API method `CreateIdentityProvider`.
4. Seu aplicativo usa AWS Cognito Identity para autenticação e armazena perfis de usuário em um pool de usuários. Para expandir a disponibilidade e facilidade de login no aplicativo, sua equipe está solicitando conselhos sobre como permitir o uso de provedores de identidade OpenID Connect (OIDC) como meio adicional de autenticar usuários e salvar as informações do perfil do usuário. Qual é a sua recomendação sobre provedores de identidade OIDC? R: This is supported, along with social and SAML based identity providers.

# AWS SWF

O Amazon SWF ajuda os desenvolvedores a criar, executar e dimensionar trabalhos em segundo plano que têm etapas paralelas ou sequenciais. Pode-se pensar no Amazon SWF como um rastreador de estado e coordenador de tarefas totalmente gerenciado.

## Características

- Criação de Tasks
- Iteração Humana
- Timeouts
- Workflow Child
- Input/Output



P: Quando devo usar o AWS Step Functions vs. Amazon Simple Workflow Service (SWF)?

Você deve considerar o uso do AWS Step Functions para todos os novos aplicativos, pois ele oferece uma abordagem mais produtiva e ágil para a coordenação de componentes de aplicação usando fluxos de trabalho visuais. Se você precisar de sinais externos para intervir em seus processos, ou você gostaria de lançar processos filhos que retornam um resultado para um pai, então você deve considerar o Amazon Simple Workflow Service (Amazon SWF). Com o Amazon SWF, em vez de gravar máquinas de estado em JSON declarativo, você escreve um programa de decisão para separar as etapas de atividade das etapas de decisão. Isso fornece controle total sobre sua lógica de orquestração, mas aumenta a complexidade do desenvolvimento de aplicativos. Você pode desenvolver programas de decisão na linguagem de programação de sua escolha ou usar o Flow Framework para utilizar construções de programação que estruturam interações assíncronas para você.

## ➤ Pontos de Atenção

You are working for a startup as its AWS Chief Architect. You are currently assigned on a project that develops an online registration platform for events, which uses Simple Workflow for complete control of your orchestration logic. A decider ingests the customer name, address, contact number, and email address while the activity workers update the customer with the status of their online application status via email. Recently, you were having problems with your online registration platform which was solved by checking the decision task of your workflow. In SWF, what is the purpose of a decision task? R: It tells the decider the state of the workflow execution.



# AWS AppSync

Acelere o desenvolvimento de aplicações com APIs GraphQL dimensionáveis

As organizações optam por construir APIs com GraphQL porque isso os ajuda a desenvolver aplicações mais rapidamente, dando aos desenvolvedores front-end a capacidade de consultar vários bancos de dados, microsserviços e APIs com um único endpoint GraphQL.

O AWS AppSync é um serviço totalmente gerenciado que facilita o desenvolvimento de APIs GraphQL ao lidar com o trabalho pesado de se conectar com segurança a fontes de dados como o AWS DynamoDB, Lambda e muito mais. Adicionar caches para melhorar o desempenho, assinaturas para oferecer suporte a atualizações em tempo real e armazenamentos de dados do lado do cliente que mantêm os clientes off-line em sincronia são igualmente fáceis. Depois de implantado, o AWS AppSync dimensiona automaticamente seu mecanismo de execução da API GraphQL para cima e para baixo para atender aos volumes de solicitação da API.

## Benefícios

Potencialize suas aplicações com os dados certos, de uma ou mais fontes de dados com uma única solicitação de rede usando GraphQL. O AWS AppSync facilita a proteção dos dados do seu aplicativo usando vários modos de autenticação simultâneos e permite que você defina segurança, armazenamento em cache e controle de acesso refinado no nível de definição de dados diretamente de seu esquema GraphQL.

Com assinaturas GraphQL gerenciadas, o AWS AppSync pode enviar atualizações de dados em tempo real por Websockets para milhões de clientes. Para aplicações Web e para dispositivos móveis, o AppSync também fornece acesso aos dados locais quando os dispositivos ficam off-line e sincronização de dados com resolução de conflitos personalizável, no momento em que estão on-line novamente.

O AWS AppSync oferece configuração, administração e manutenção de API GraphQL totalmente gerenciadas, com infraestrutura sem servidor de alta disponibilidade integrada. Crie uma API GraphQL em segundos por meio da AWS CLI, do console, Amplify CLI ou CloudFormation. E monitore e adquira facilmente percepções via CloudWatch e X-Ray para uma operação perfeita.

